Visualization of Reddit using Text Mining and Hierarchical Clustering

Per Blåwiik

Abstract—Based on the amount of available data from text based resources, it is relevant to explore potential usages for it. The purpose of this report was to present how to visualize and understand the website Reddit by utilizing common text mining methods, hierarchical clustering and force-directed graph algorithms. The fundamental idea was to use the most frequent words as basis for measuring similarities and subjectively interpret the content of subreddits. Using a force graph as a visual representation turned out to be intuitive, aesthetically pleasing and practical. Based on subjective analysis, the most frequent words deemed to be suitable both for clustering and understanding written text data.

Index Terms—Text mining, Hierarchical clustering, Force graph, Cosine similarity, Information visualization.

1 Introduction

Text mining, also referred as *text data mining*, can be briefly explained as a process where features or patterns are extracted from text based resources [1]. Examples of text based resources are the written content of websites such as online forums, customer reviews and articles. In the current age of social media, these types of resources are overflowing with available data and therefore it is relevant to explore the possible usages for it.

In order to understand abstract data and discover patterns, a graphical visualization tool is often necessary. Such a tool can include several steps: preparation of data by extracting important features, clustering data objects based on the features and visualize them in a graphical user interface by drawing various types of charts.

1.1 Problem

In this report, the implementation of a tool for visualizing the public online forum *Reddit* is explained. The content of reddit consists of posts made by the members of the website. A post consists of written text, links or images and can be up or down voted by other members. All posts belong to user-created topics called *subreddits*.

The purpose of the visualization tool is to help understand the contents of subreddits and find similarities between them. The following research questions were the basis for the project:

- Can the most frequent words in an arbitrary subreddit be used as an accurate description of the content of that subreddit?
- Can similarities between subreddits be identified when comparing their most frequent words? Can these words be used to represent a subreddit in order to cluster similar subreddits?
- Are there any correlations between the most frequent words and the given average scores of the texts where the words appeared?

The purpose of this report was to present methods that can help to answer the research questions with a visual representation of Reddit.

2 Background and Related Work

A common application of text mining is the matching of search phrases (queries) against documents in a large database, based on relevance. For instance, if a database consists of medical journals, a doctor can form a query of keywords relevant to a particular syndrome. The query is then matched against the abstracts of all papers in the database and the system returns a list of documents ordered by a relevance ranking.

• *Per Blåwiik is with Linköping University, Sweden. E-mail: persj146@student.liu.se.*

The query matching method is closely related to the second research question presented in Section 1.1: identify similarities between subreddits based on their most frequent words. For solving this problem, all posts in a subreddit must first be merged into a large text and then the words are counted. The top N most frequent words are then extracted into a vocabulary to represent that subreddit. Similarly to query matching, the subreddit vocabularies can be used to compute the relevance between them. Note that for query matching, a query is compared to a document collection. In the presented case, however, each document is compared to the document collection.

3 Data

The data used for this report was derived from a dataset containing about 1.7 billion posts released by Reddit. All posts from the month of May 2015 (29.6 GB of data) is available online at *Kaggle*. Each post in the original dataset contains 22 features, however, only three of them were needed to answer the research questions: *Subreddit*, *Body* and *Score*. Subreddit is the name of the subreddit which the post belongs to; body contains the written text of the post; score is the number of up votes minus the number of down votes given to the post. An example of the raw format of the data is seen in Figure 1.

Post		
Subreddit	Body	Score
soccer	Zlatan knows how to kick a ball!	333
AskReddit	I'm a guy and I had no idea this was a thing guys did.	-10
music	Did well on my quiz today, am now eating ice cream. Good day.	25

Fig. 1. The raw format of the data used for visualizing Reddit.

In the final implementation (before text preprocessing), 200 Megabytes of data were used. This corresponds to about 100 different subreddits, where each subreddit contains between 1000 - 20000 number of posts. These subreddits were subjectively chosen to cover a broad range of distinguished topics. The motivation for doing this was to make it easier to assess the success of clustering by relevance.

4 Method

In this section, the methods that were used for preprocessing, formatting, clustering and visualizing the data are presented.

4.1 Text Preprocessing

With the reduced dataset of 200 MB described in Section 3, several preprocessing steps were performed before the subreddits could be

used for interpretation, measuring and clustering.

In the field of information retrieval, words that contributes to describe the content of a document is called *terms* [1]. In this case, a subreddit can be interpreted as a single document of which the content consists of the top N most frequent terms linked to the subreddit.

Before the terms are counted, the text requires filtering. The filtering was done by converting all characters to lower case, expand contractions and removing punctuations, special characters, *stop words* and short words (less than three characters). Stop words are words that appear in most documents and can therefore be considered as noise. For this project a stop list of 759 words were used. The reason for removing short words were the same as for using stop words.

After the filtering, the remaining terms of all subreddits were counted and separate vocabulary lists were created. The top N most frequent terms were then extracted to compose a document to represent the corresponding subreddit.

4.2 Vector Space Model

The concept of the *vector space model* is to create a *term-document matrix* $A \in \Re^{MxN}$, where each document is represented as a column vector, M is the number of documents and N is the number of unique terms in the document collection [1]. Each element (row position) of a column vector represents the term count in that document. Consequently, by creating a term-document matrix, all documents (subreddits) share the same dimension and have numerically calculable values (unlike text strings).

In addition to counting the term occurrences, a term weighting scheme called *TFIDF* (term frequency-inverse document frequency) was applied. The idea is to use a logarithmic scaled weight to make rare terms more important than common terms (found in most documents). Consider the term document matrix A, then, each element is given by Equation (1).

$$a_{ij} = f_{ij} log(n/n_i) \tag{1}$$

The term frequency f_{ij} is the number of times term i occur in document j, n is the total number of documents and n_i is the number of documents that contain term i.

The TFIDF method was used because it makes certain terms more relevant to certain documents which contributes to distinguish them.

4.3 Cosine Similarity

The most important part of the query matching problem presented in Section 2 is to measure the relevance between the query and the documents. A common method in information retrieval is to use the *cosine similarity*: two documents are considered to be similar if the angle between them is smaller than a predefined tolerance value $t \in [0,1]$. The cosine similarity is found by solving the dot product equation for cosine and compare it to a tolerance value, see Equation (2):

$$cos(\theta(a_i, a_j)) = \frac{a_i^T a_j}{\|a_i\|_2 \|a_j\|_2} > t \tag{2}$$

where a_i and a_j are column vectors in the term-document matrix A introduced in Section 4.2, and t is a predefined tolerance value. Cosine similarity is used mainly because the norms (length) of the documents do not affect the angle between them.

4.4 Hierarchical Clustering

For the strategy of clustering similar subreddits, *agglomerative* hierarchical clustering analysis (HCA) was used. Initially, all objects are treated as separate clusters and are then iteratively grouped into a hierarchy of clusters [3]. The grouping is achieved by calculating the distances between all clusters and merge the two that are the closest. When the clustering is complete a *dendrogram* is usually created to display the hierarchical relationships between the clusters. From analysing the dendogram in Figure 2, a suitable number of clusters for the final cluster labeling was found.

For the linkage criteria, that is, from where the distance is computed, *Ward's minimum variance method* was used. In this method, the strive is to minimize the total variance within each cluster [4]. The initial cluster variances (for singletons) are defined as the squared Euclidean distances in Equation (3).

$$d_{ij} = \|a_i - a_j\|_2^2 \tag{3}$$

In order to use the Euclidean norm, a distance matrix was first computed based on the cosine similarities between the documents (see Section 4.3). The similarities were then used as features for each subreddit to determine their euclidean distances from each other. Ward's linkage criterion was used because it is considered to be the sensible default method and it also proved to work well for the Reddit data.

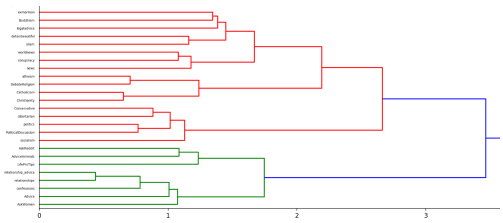

Fig. 2. A part of the dendogram, generated by hierarchical clustering, that was used to find a suitable number of clusters to use for the subreddits.

4.5 Force Graph

To visually represent the relationships between subreddit, a *force-directed graph drawing algorithm* was used. The method is used for computing layouts for undirected linked graphs, based on physical simulations (typically spring force systems). The idea is to initially place nodes on random positions, treat the graph as a spring system and then look for stable configurations [2]. Once the system is stable, the nodes has been positioned in a aesthetically pleasing and symmetrical manner.

The forces (weighted links) between nodes (subreddits) used in the implementation were directly based on their mutual cosine similarities explained in Section 4.3. A tolerance value was used to determine if a link between two nodes should exists.

The force graph was chosen because it offered a clear and intuitive overview of the subreddits and also worked well with the use of colours based on the clustering presented in Section 4.4.

5 Implementation

The implementation of the visualization tool can be divided into three subsystems: *preprocessing*, *back-end* and *front-end*.

The preprocessing system was developed with the programming language *Python* and is the most computationally heavy part. Python was used because there exists a large number of packages dedicated for data mining tasks for it. This system performs the following tasks:

- the selection of data from the original database and text preprocessing of the selected data (explained in Section 4.1);
- the transformation of text data into the vector space model presented in Section 4.2;
- the calculation of cosine similarities, hierarchical clustering and the creation of an undirected linked graph system for the force graph explained in Sections 4.3, 4.4 and 4.5.

Finally, all processed data is saved to a local file in *JSON* format to be accessed directly by the back-end and front-end systems. Note that this system is only runned offline since the tasks are very time consuming.

The back-end system is the server side of the tool and was implemented for handling filtering applied through the front-end. However, the only task in the final product was to merge word count lists that represents subreddits. Python was used to develop this system because it exists a lot of documentation for server-client communication as well as for applied data mining.

The front-end system is the graphical user interface where users can interact with and explore the dataset. The force graph is the main tool for navigating this system. It offers the user to pan and zoom the view, and select subreddit nodes to understand their content. This system was developed as a *HTML* page using *JavaScript* and the library *D3*. A web browser based platform was used because it is supported by most computer systems. An additional advantage was that there were numerous of available examples for visualizing most types of data in various approaches using D3.

6 Results

The final product was a visualization tool available for the most popular web browsers. The graphical user interface (front-end) is running on the client side and requires a connection to the server (back-end) to work. The result is an overview of 110 different subreddits laid out in a force graph where the nodes are coloured by a group id (generated from the hierarchical clustering). The most suitable number of clusters were subjectively found to be 12. The nodes spatial positioning is based on the direct similarities (cosine similarity) between the subreddits. A tolerance value of 0.09 was used to determine if two subreddits should be linked together. Each subreddit is approximated and represented by the top 50 most frequent words found in the posts belonging to the subreddit.

The main way of navigating the application is to interact with the force graph found in the center of the screen, see Figure 3. The user can pan and zoom the view of the force graph as well as selecting one or multiple nodes with the mouse. When selecting nodes, a list of the top 50 most frequent words is shown in a panel to the right side of the screen, see Figure 4.

The application responds instantaneously when selecting multiple subreddits since most of the computations were performed offline. All research questions presented in Section 1.1 could be answered by using the tool.

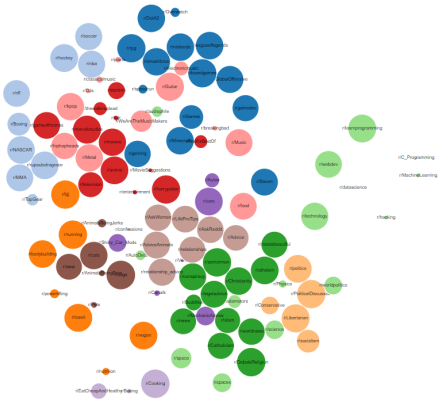

Fig. 3. The force graph visualizing all subreddits in the dataset. The node colours are based on hierarchical clustering, meaning that same coloured subreddits are related.

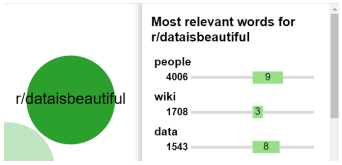

Fig. 4. A list of the top 50 most frequent words of the subreddit called *dataisbeautiful*. To the left below the words are the number of occurrences and to the right the average score.

7 Evaluation

To determine if the visualization tool was intuitive and practical, an evaluation scheme was compiled. The aim of the evaluation was: to find out if the application was easy to understand and navigate, and, to learn if the visual representation suited the data.

7.1 Evaluation Scheme

Three evaluation methods were used: observation, think-out-loud and interview. By observing the actions of the participant when navigating the application, subconscious issues with the user interface can be revealed. When using the think-out-loud method, the interviewer can follow the thought process of the participant and discover subjective assumptions and frustration triggers. Finally, in the end, the participant can answer questions with their own words to highlight certain problems and offer suggestions for improving the system.

At the beginning of the session, the interviewer explained the dataset and the visual representation to the participant. Once the participant understood what they saw, they were given four tasks to complete. All tasks were aimed to force the participant to use the entire functionality of the application as well as to understand the visual representation of the data. While solving the tasks, the participant were asked to think-out-loud. The interviewer's role was to observe, take notes and intervene if the participant should get stuck. The participants had at all time access to the tasks in written form.

When the tasks were complete, the participants were given five questions to answer on a paper, asking: the purpose of the application, if anything was unclear or difficult, opinions on expected behaviour or responsiveness, and what they had learned from the data. At the end of the question form, they were also asked to make two statements on a scale from 1 to 4. An even scale was used to prevent neutral statements.

7.2 Evaluation Results

The evaluation was performed by five participants with no or little prior knowledge of Reddit. Note that this was just a coincident and not planned. The findings of the the evaluation was: the interface was easy to learn, the visual representation of the data was intuitive, the purpose of the tool was clear and the application was missing expected *quality-of-life* functionalities (e.g. search bar, single select multiple nodes).

8 Conclusions and Future Work

All three of the research questions presented in Section 1.1 could, based on subjective analysis, be answered by using the final product: the most frequent words could describe the content of subreddits, similarities between subreddits was in many cases identified by using their most frequent words, the clustering of similar subreddits was successful and no correlation could be found between a words average score and its number of occurrences.

The force graph proved to be an intuitive visualization of the relevance between different subreddits (based on the nodes position and color) as well as offered a clear overview of the dataset. Panning and zooming the graph was fun and useful tools for exploring and understanding the dataset.

Some potential improvements for future work are to add a cluster bar to make it possible to select entire clusters, add a search bar for the force graph to select subreddits based on a search phrase and add a search bar for the list of the most frequent words.

References

[1] L. Eldén. *Matrix Methods in Data Mining and Pattern Recognition*, volume 2. SIAM, Philadelphia, PA, USA, 2019.

[2] S. G. Kobourov. Spring embedders and force-directed graph drawing algorithms. January 2012.

[3] L. Rokach and O. Maimon. Clustering methods. In *Data Mining and Knowledge Discovery Handbook*, pages 321–352, Boston, MA, USA, 2005. Springer.

[4] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, pages 236–244, 1963.